Allied Research

CASE FILE COPY

ALLIED RESEARCH ASSOCIATES, INC

BOSTON, MASSACHUSETTS

Document No. ARA-T-9173-1

12 February 1963

Prepared for

National Aeronautics and Space Administration
Goddard Space Flight Center
Aeronomy and Meteorology Division
Greenbelt, Maryland

by

Christopher Dean
Arnold H. Glaser

TECHNICAL NOTE NO. 2
CONTRACT NAS 5-1204

COMPUTER REQUIREMENTS FOR PRINCIPAL ARACON PROGRAMS IN FUTURE TIROS OPERATIONS

ARACON LABORATORIES
a division of Allied Research Associates, Inc.
Virginia Road, Concord, Mass.

15/30

FOREWORD

ARACON Laboratories Division of Allied Research Associates, Inc. has been operating computer facilities at the NASA Wallops Station and Point Mugu CDA stations in support of TIROS operations. The support involves orbit-by-orbit determination and prediction of satellite camera attitude and the use of these attitudes for the computation and automatic drawing of latitude - longitude perspective overlay grids for geographic referencing of the TIROS pictures.

This report provides basic documentation of the two principal programs involved -- the TIROS Sequence of Picture Grids (SPG) latitude - longitude grid drawing program and the TIROS Horizon Sensor data reduction programs (H-1 and H-5). With the exception of the MGAP program, all the other principal programs used at the CDA stations use much the same basic geometry and similar formulas. Input-output is tailored to the specific requirements.

The motivation for this report is the necessity for basic information on program nature to permit intelligent selection of a successor computer to the Bendix G-15 D units now in operation at the CDA stations. It is probable that their capacity will be exceeded by the data from the large number of TIROS satellites now planned for early launch. The probable use of eccentric orbits and unconventional camera systems will entail extensive reprogramming in any event, so that the computer change should be made at this time.

TABLE OF CONTENTS

SECTION 1

TIROS GRID PROGRAM

1. Introductory

1.1 Introduction

The current Allied Research latitude-longitude grid program was developed in a crash program of a few weeks' duration and was delivered for use at TIROS III launch. For maximum programming speed it was constructed in sections by different programmers working virtually independently, and normally the first codings which worked were accepted on the spot. Additional linkage programs were inserted during final assembly to compensate for the contrasting coding practices that had evolved.

In addition to furnishing a quick expedient for TIROS III, this program was intended to supply a first trial of concepts to be used in the final full reprogramming that was anticipated in the event that operations continued with further satellites. Unfortunately, the temporary expedient has served too well, and the need for a major final effort of programming and documentation has not been apparent until the present.

The present report includes background considerations, basic geometry and mathematics used, and mathematical flow chart for the grid calculations. Coding details, program bookkeeping, and both internal and external service routines are omitted. However, details of program operation are included.

1.2 Origin, Motivation and Basic Concepts

The program section which draws a single grid is based on the original Allied Research grid program that was developed under Air Force Contract AF 19(604)-5581 (Reference 1, Appendix A) and which was tested at Pt. Mugu during operations with TIROS II under U.S. Weather Bureau Contract Cwb 10023. Although recoded for better efficiency and to fit slightly altered geometry, fundamental changes were avoided in order to insure meeting the launch date for TIROS III.

The test with TIROS II had made it abundantly clear that real-time operational use would be feasible only with the capability for producing entire sequences of grids automatically. This meant automation of the input data for the individual grids, and also very reliable built-in tests to insure that understandable and usable outputs would be produced steadily throughout an hour or more of unattended operation. The major programming effort on the grid program for TIROS III was to implement these added features while maintaining a reasonable speed of operation.

A major new section in the grid program calculates satellite ephemerides, continuing through satellite orientation, to produce all of the constants needed for the object-image transformation in the camera, and including the derivation of various geographical locations important for the grid interpretation. The repetitive type-in of common data was virtually eliminated by storing satellite orbit elements and sun ephemerides on magnetic tape, and the necessary tape handling and table search routines were adapted and fitted into the overall program. Separate programs were written to control the preparation and storage of this data on tape; those programs will be described elsewhere.

To streamline the sections most often repeated, and to permit automatic recycling of the relatively large total program needed at present for each grid, those sections are stored on a second magnetic tape. An indexed executive program was devised to control the reading of program commands from this tape; it permits full buffering of both reading and file-code searching, while providing at the same time for re-reading any block which shows a check-sum error. Since most of the computer memory is now full most of the time, it is usually not possible to use the standard Bendix Program Preparation Routine with it. Therefore, it was necessary to devise more compact service routines to enter commands and to facilitate debugging and program modification.

Consideration was given to common magnetic tape storage and consolidation with the other programs which were being developed at the same time for TIROS III launch (MGAP, ODAM, H-1, etc.), since they require portions of the same input data and of the same ephemerides. However, it seemed unwise to attempt this level of integration during the crash program shortly before TIROS III launch.

A major consideration throughout all of the programs was to insure efficient use of the operator's time in the actual day-to-day applications. Thus the format and units of typed-in information were chosen to coincide as nearly as possible with those which have become the traditional daily "lingo" at the TIROS readout sites. Furthermore, the type-ins of instruction codes and data to the programs is grouped as much as feasible into introductory portions of the program, after which the machine is left to operate unattended for as long as possible.

2. Formulation

2.1 Basic Geometry: Description and Calculations

2.1.1 Orbit Elements

The grid program is designed to use orbit elements as they are received on teletype at the TIROS read-out stations. Typically, a message is received about once per week giving the semi-major axis, eccentricity, inclination, argument of perigee, and right ascension of ascending node at a recent epoch, as well as the first time derivative of the last two quantities. The position of the satellite at epoch is given by its mean anomaly. Certain other quantities are included in the message, of which the present program uses the anomalistic period and its first time derivative. Whenever one of these messages is received, these quantities are stored on magnetic tape, to be read in by the computer whenever needed.

The orbit elements at a later epoch are calculated by linear extrapolation, i.e., for each quantity Q,

$$Q = Q_o + (t - t_o)\, DQ \qquad (1)$$

where Q_o is the value received in the message, and DQ its rate of change. The epoch of the message is t_o and the new epoch is t, such that $t - t_o$ is elapsed time (in days and fraction of a day, if DQ is used as received).

For each sequence of grids to be drawn, the time and longitude of an adjacent northbound equator crossing is typed in, using the values given in the complete list of these data which is also received via teletype. In principle, these also could be calculated. However, in effect the extrapolation for these quantities involves the total rotations of the earth and total revolutions of the satellite around its orbit since epoch. These are not small quantities, so that the desired accuracy would require relatively high precision in the extrapolation.

The longitude of the ascending node, θ, is now found from

$$\theta = \theta_1 + (D\Omega - DS)\,(t - t_1) \qquad (2)$$

where θ_1 and t_1 are the longitude and time of the northbound equator crossing as typed in, DΩ is the time derivative of right ascension of ascending node, DS is the sidereal angular velocity of the earth, and t is the time for which θ is being calculated. The alternative possibility of deriving this quantity by interpolation between NASA values for two adjacent northbound crossings was discarded to reduce the number of typed inputs (all the other quantities are available to the computer from the magnetic tape storage).

2.1.2 Coordinate Systems

The preceding paragraphs use the usual geocentric equatorial coordinate system: unit vectors $\underline{i}$ toward vernal equinox, $\underline{k}$ toward north celestial pole, and $\underline{j}$ to form a right-handed triad from $\underline{i}$, $\underline{j}$, $\underline{k}$. The right ascension and declination are conventional polar coordinates referred to this reference frame.

We shall also refer to other standard coordinate systems, for much of the computation in the grid program simply consists of repeated transformations between them. We describe these coordinates by specifying the orientation of the right-handed triad, $\underline{i}$, $\underline{j}$, $\underline{k}$ required in each case:

Geographic coordinates are geocentric, $\underline{i}$ is in the Greenwich meridian, $\underline{k}$ is north; we use north geocentric latitude and east longitude referred to these axes.

Orbital Nodal coordinates are geocentric, $\underline{i}$ points to the ascending node, $\underline{k}$ is the positive normal to the plane of the orbit. The effective longitude in this system is known as argument (or even "argument of latitude"); the effective latitude, b, has no common name.

Local Orbital coordinates are geocentric, $\underline{k}$ is the positive normal to the orbit plane, $\underline{i}$ points toward the satellite.

Local Geographic coordinates are geocentric, $\underline{i}$ points to the satellite, $\underline{k}$ points north as seen from the satellite.

Topocentric orbital or geographic coordinates are oriented just like the corresponding local coordinates, but the origin is taken at the satellite (at the front nodal point of the satellite camera lens).

Although much description uses angles, the transformations are all treated in the grid program in cartesian terms, as matrices multiplying vectors; the trigonometry is simply collected into cartesian-polar conversions and into derivation of matrix elements. We shall derive any rotation as a sequence of simple rotations of the triad of unit vectors, in each of which one axis remains fixed. We define $[I(\phi)]$ to signify the matrix for one of these rotations, where I = 1, 2, or 3 specifies that the rotation was around $\underline{i}$, $\underline{j}$, or $\underline{k}$ respectively; ϕ is the amount of rotation. It can be seen that $[I(\phi)]^{-1} = [I(-\phi)]$. For reference, we list the formulas for the simple rotations:

$$[1(\phi)] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \quad (3a)$$

$$[2(\phi)] = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \quad (3b)$$

$$[3(\phi)] = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3c)$$

2.1.3 Satellite Position

In the present program, the satellite position at picture time is computed using conventional approximations for orbits of small eccentricity. To first order in eccentricity, we have

$$r/a \doteq 1 - e \cos M \quad (4a)$$

$$v \doteq M + 2e \sin M \text{ (radians)} \quad (4b)$$

where a is the semimajor axis and e the eccentricity, and where M is the mean anomaly, v the true anomaly and r the distance from the center of the earth to the satellite. These formulas introduce errors the order of e^2, the first terms omitted. A variation of TIROS altitude of 100 miles above and below programmed height corresponds to a major launch error which produced an eccentricity the order of $e \doteq .025$; this case was realized with TIROS V. For this case Equation (4a) produces errors the order of $ae^2 \doteq 4000 \times .000625$, or 2.5 miles in height. Equation (4b) produces errors the order of $e^2 = .000625$ radians, or roughly .04 degrees in anomaly; this in turn corresponds to about 2.7 miles in position.

Clearly the errors just quoted are well within the requirements for operational analysis of TIROS pictures. However, the present program calculates seriously erroneous positions for a truly elliptical orbit such as the suggested TIROS experiment in which height would vary between 300 and 3000 miles. The approximations actually do not save enough operating time or program storage space to justify this limitation, and it is suggested that this be rewritten using conventional calculations without mathematical approximations.

In the present program the orbit is also specified approximately. The basic time reference for all calculations concerning one sequence of picture grids is taken as the northbound equator crossing for which time and longitude were typed in. Therefore, the orbit elements are extrapolated to this epoch using Equation 1, and those values are used throughout that sequence of grids without calculating the true additional small changes. The argument of perigee changes by only a fraction of a degree through a sequence, and the satellite position errors generated by ignoring this change are only the order of eccentricity times the angle error. In the program the right ascension of ascending node is used explicitly only in calculating sun position relative to the satellite, and the fraction of a degree by which it changes is of no consequence.

In calculating the satellite geographic position it is necessary to compensate for the motion of perigee which is ignored above. This is done by using the nodal period instead of the anomalistic period in defining the effective values of mean anomaly, M, to be used in Equations (4a), (4b):

$$M = 2\pi (t - t_p) / T_n \quad \text{(radians)} \qquad (5)$$

where $(t - t_p)$ is time elapsed since perigee, and T_n is the nodal period, the same units being used for both.

The extrapolated orbit elements provide the argument of perigee directly, whereas the time of perigee is required for the mean anomaly; hence the time must be calculated. The program accomplishes this as soon as the orbit elements at northbound equator crossing have been determined: the argument of perigee is inserted for v in Equation (4b), the resulting transcendental Kepler's equation is solved for M, and finally $t_p - t_1 = (MT_n) / 2\pi$, where t_1 is the time of northbound equator crossing, t_p is time of perigee, and T_n is nodal period.

From the foregoing calculations, the argument of the satellite, L, is obtained as

$$L = w + v \tag{6}$$

where w is the argument of perigee at the time of northbound equator crossing, and v is true anomaly of satellite according to the approximation described above. The inclination is known, of course. Finally, we determine the longitude of ascending node at the actual time in question (i.e., picture time) from Equation (2), and the position of the satellite is completely specified relative to the earth, in orbital nodal coordinates.

Conversion to geographical coordinates requires a negative rotation about ascending node by the inclination, i, followed by a negative rotation about north by the longitude of the ascending node, Θ. Dividing out the radial distance, r, we deal with the subpoint on a spherical earth of unit radius:

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = [3\,(-\Theta)]\;[1\,(-i)] \begin{Bmatrix} \cos L \\ \sin L \\ 0 \end{Bmatrix} \tag{7}$$

where x, y, z are cartesian components relative to the geographical system, such that the longitude β and the geocentric latitude γ are given by

$$\begin{aligned} \gamma &= \arcsin(z) \quad -90^\circ \leq \gamma \leq 90^\circ \\ \sin\beta &= y/\cos\gamma \\ \cos\beta &= x/\cos\gamma \end{aligned} \tag{8}$$

2.1.4 Description of Directions

A unit vector may be translated at will without affecting the direction which it specifies. Thus we may visualize it as being affixed to the origin of any coordinates that are convenient, and describe it by the corresponding cartesian or polar coordinates. At various times the present program uses directions specified in the usual geocentric equatorial system, the orbital nodal system, and in both local systems (see section 2.1.2; note that each local and its associated topocentric system are equivalent for describing a direction).

The conversion from equatorial to orbital nodal systems is performed as a standard transformation, as described earlier. This conversion is needed, for example, for the sun ephemeris. We write the original cartesian components of the direction explicitly as functions of its right ascension, α, and its declination, δ. The transformation consists of a rotation about north by Ω, the right ascension of ascending node, followed by a rotation about the node by i, the inclination of the orbit. The transformed angles L, effective longitude (argument), and b, effective latitude relative to the orbital nodal reference frame, are written below as functions of the transformed cartesian components (x, y, z):

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = [1\ (i)] \cdot [3\ (\Omega)] \cdot \begin{Bmatrix} \cos\delta \cos\alpha \\ \cos\delta \sin\alpha \\ \sin\delta \end{Bmatrix} \quad (9)$$

$$\begin{aligned} b &= \arcsin z \\ \cos L &= x/\cos b \\ \sin L &= y/\cos b \end{aligned} \quad (10)$$

The conversion from orbital nodal to the local systems is written in a special way for the directions of the sun and the satellite principal axis. Their local orbital values are obtained by a single rotation about the normal to the orbit, by the argument of the satellite. If we use primes for nodal and double primes for local coordinates of the direction, and leave the satellite argument unprimed, we have

$$\begin{aligned} b'' &= b' \\ L'' &= L' - L, \end{aligned} \quad (11)$$

so that the cartesian coordinates in the local orbital system may be written directly in the form

$$\begin{aligned} x'' &= \cos b'' \cos L'' = \cos b' \cos (L' - L) \\ y'' &= \cos b'' \sin L'' = \cos b' \sin (L' - L) \\ z'' &= \sin b'' = \sin b' . \end{aligned} \quad (12)$$

The principal axis is to be described by an azimuth angle, ξ, and nadir angle, n. These are defined in topocentric coordinates (origin at front nodal point of camera lens), and concern the line segment between the camera lens and the principal point in object space (on surface of earth or celestial sphere). To fit into the present discussion, let us consider the positive direction on this line to be from the origin (lens) to the principal point, although either sense of direction may be understood in subsequent applications. The nadir angle is measured between this direction and the local vertical, "down," i.e., the negative $\underline{i}$ unit vector of our local reference triad. Thus

$$n = \arccos(-x'') \tag{13}$$

$$\text{or} \quad \cos(n) = -\cos(b')\cos(L' - L) \quad 0 \leq n \leq 180^\circ$$

The azimuth angle is readily visualized in terms of the projection of the principal axis onto the plane tangent to earth at the subpoint: an observer in the satellite sees a counterclockwise rotation of this projection for a positive azimuth angle. In the orbital coordinates the forward orbit direction is taken as reference, so that

$$\begin{aligned} \cos\xi &= y''/(1 - x''^2)^{1/2} \\ &= \cos(b')\sin(L' - L)/\sin(n) \\ \sin\xi &= Z''/(1 - x''^2)^{1/2} \\ &= \sin(b')/\sin(n) \\ \text{or} \quad \operatorname{ctg}(\xi) &= \operatorname{ctg}(b')\sin(L' - L) \end{aligned} \tag{14}$$

The variation of these directions relative to the local coordinates is much faster than their variation relative to either space or the orbit. In effect, L is the independent variable, and L' is a slowly changing system parameter. Furthermore, it is obvious from Equation (13) that the minimum nadir angle occurs for $L = L' + 180^\circ$, and that it has the value $n = b'$. Consequently, we rewrite Equations (13), (14) in the form

$$\cos(n) = \cos(n_o)\cos(L - L_o) \qquad 0 \leq n \leq 180^\circ \tag{15a}$$

$$\operatorname{ctg}(\xi) = \operatorname{ctg}(n_o)\sin(L - L_o) \quad \text{(sign of } \xi \text{ is same as sign of } n_o) \tag{15b}$$

where we introduce the convenient parameters n_o, the minimum nadir angle, and L_o, the value of argument of satellite at which the minimum nadir angle occurs. The formula for ξ by itself does not completely specify the quadrant as it is written. However, we note that if b' is positive, we should have $0 \leq \xi \leq 180^{\circ}$ whereas for negative b' we have $-180^{\circ} \leq \xi \leq 0$. Thus it is convenient to include the sign of b' in the parameter n_o, i.e., $n_o = b'$, and add to Equations (15a), (15b) the specification that the azimuth angle shall have the sign of n_o.

We emphasize that the foregoing refers to the orbital azimuth angle. The geographical azimuth is obtained by subtracting the orbital azimuth of the "north" vector, a quantity derived below.

The direction from the earth to the sun is found in right ascension and declination from standard published ephemerides, which have been read into magnetic tape storage for ready access by the computer. For the purpose of the present program it is useful to deal with the direction of the shadow of the satellite, which is simply the opposite of the above, so that we use

$$\begin{aligned} \alpha \text{ shadow} &= \alpha_{\text{sun}} + 180^{\circ} \\ \delta \text{ shadow} &= -\delta_{\text{sun}} \end{aligned} \qquad (16)$$

This is converted to orbital nodal coordinates by Equations (9), (10) for the time of the northbound equator crossing, and n_o^{sun} and L_o^{sun} determined, after which the azimuth and nadir angles of the shadow are computed from Equations (15a), (15b). Ample accuracy for our work is provided by determining n_o^{sun} and L_o^{sun} for successive northbound equator crossings and then using linear interpolation for actual picture times.

The unit vector pointing toward North is readily expressed in orbital nodal coordinates. It is tipped from the positive normal to the orbit by the amount of the inclination, so that $n_o^{\text{north}} = (90 - i)$ degrees. The tip of the vector lies closest to the orbit along $L = 90^{\circ}$, so that the argument of minimum nadir angle for north is $L_o = 270^{\circ}$. The orbital azimuth of North is found by using these values with Equation (15b), or (depending on programming choices) by the specialized formula

$$\begin{aligned} &\text{ctg}\,(\xi^{\text{north}}) = \tan\,(i)\,\cos\,L \\ &\xi \text{ has the sign of } (90 - i) \text{ degrees} \end{aligned} \qquad (17)$$

2.1.5 Camera Orientation

In the grid program it is necessary to consider the principal axis as being precisely parallel to the satellite spin axis. It is the spin axis which is predicted from satellite dynamics, so that the grid derivation must accept it as basic input information. Ideally, the optical axis of the TIROS camera parallels the spin axis, but, of course, there is always some discrepancy in practice. The discrepancy is determined during calibration and is most easily handled by including it as part of the overall lens distortion which is determined at that time.

For each sequence of picture grids, the grid program accepts a typed input of the parameters n_o and L_o for the principal axis, at the time of the reference northbound equator crossing. An input of the first time derivatives is also permitted, although the changes are usually small enough relative to the precision with which the attitude is known to ignore this detail. For each grid in the sequence the principal axis nadir and azimuth angles are computed using Equations (15a), (15b).

The roll angle of the satellite (and camera) around the principal axis should also be incorporated in the grid program. This is necessary in order to permit allowance for non-radial camera distortions (such as the misalignment mentioned above) and for TV distortions. The roll is measured between an arbitrary fiducial direction on the satellite and an arbitrary reference direction in space. In the present program the space reference is the plane containing the satellite, the center of the earth, and the camera principal point, since this reference emerges naturally during the transformation from geography to image space, as treated in the next section. If the roll were given in other terms, such as the sun sensors were intended to provide, that input would simply be converted to the standard earth-center reference before use by the program.

In the present program the roll is fixed at zero (i.e., ignored). This came about because the present program was hurriedly adapted from a simpler single-grid program. This feature of the program has escaped repair because some grids are prepared well before the pictures are received, so that the roll angle cannot be known, and because the unreliability of the sun sensor information makes it fairly hopeless to deal with roll in any operational TIROS work at present. However, the sun sensors should work, and the program should be able to provide highly accurate second-round grids which include non-radial distortion corrections. The changes required are quite minor, and the details are included in the derivations in following sections.

2.2 Basic Image Calculations

2.2.1 Object-Image Transformation

The formation of an image is described by reference to the special camera-based coordinate system of unit vectors $\underline{i}$, $\underline{j}$, $\underline{k}$ which is shown in Figure 1. The origin is at the lens, and the vector $\underline{i}$ points toward the focal plane along the principal axis, so that it is normal to the focal plane for an ideally aligned system.

The location of a point in object space is described by coordinates (a, b, c) relative to these axes. With zero distortion, i.e., with the idealized geometry of a pin-hole camera, its image lies at the point with coordinates (f, b', c'), where f is the focal length of the lens, and where

$$\begin{aligned} b' &= fb/a \\ c' &= fc/a \end{aligned} \tag{18}$$

In all subsequent discussions f will be considered unity; its effect is absorbed into the overall factor which determines the scale of the output grid. The method of dealing with image distortions will be described later.

Initially, we describe an object location by latitude, ψ, and longitude, ρ, east latitude and north longitude being chosen as positive. The required transformation can be visualized as a sequence of rotations and a translation of the reference triad of unit vectors into the camera-based position described above, from an initial alignment along the standard position described above, from an initial alignment along the standard geographical reference frame (see section 2.1.2). The operations on the reference triad occur in the following sequence:

1. Rotation around $\underline{k}$ (north pole) by the east longitude of the satellite subpoint β : this places $\underline{i}$ at the subpoint longitude, although still along the equator;

2. Rotation around $-\underline{j}$ in its new location, by the north latitude of

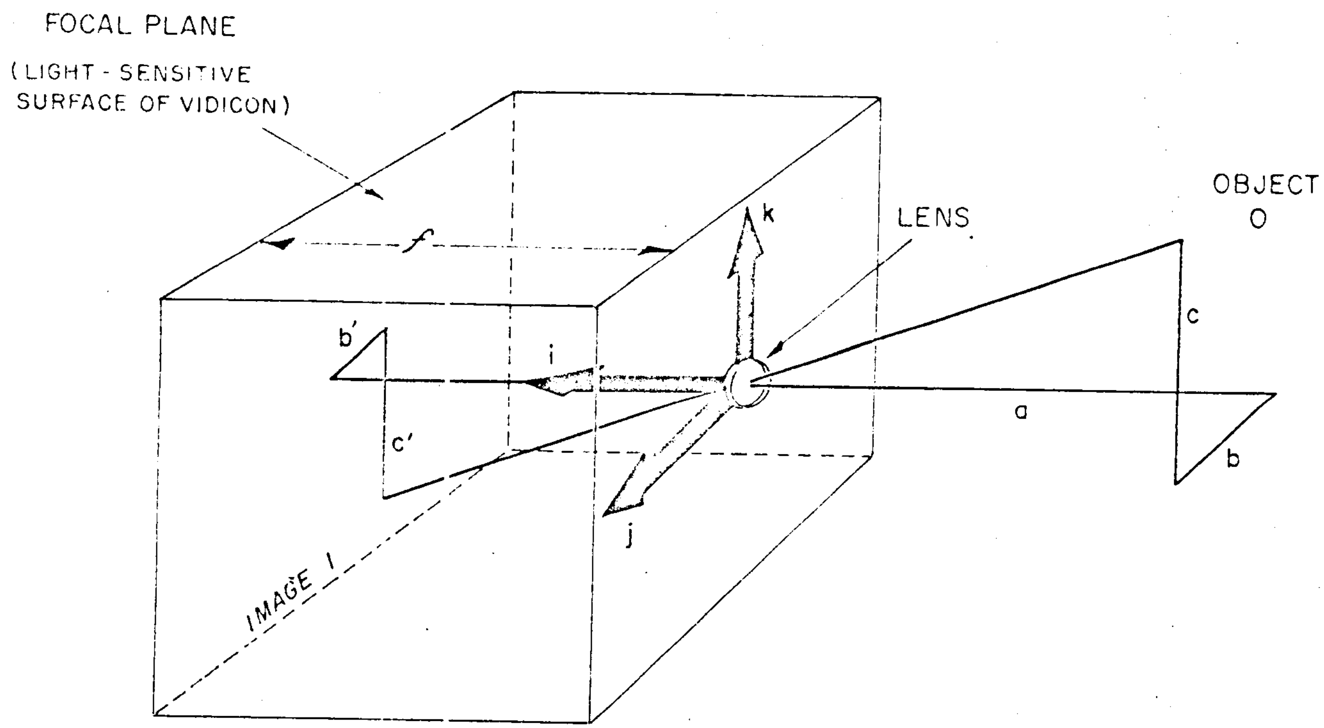

Figure 1 - Object-Image Relationship for no Distortion

the subpoint γ : this places $\underline{i}$ through the satellite subpoint itself, while $-\underline{j}$ remains along the equator, and the $\underline{i}$, $\underline{k}$, plane contains the north pole;

3. Translation parallel to $\underline{i}$ by the radius vector of the satellite from the center of the earth r: this places the origin of coordinates at the lens;

4. Rotation around $\underline{i}$ (satellite radius vector) by the geographical azimuth angle of the principal axis ξ : this places $\underline{k}$ parallel to the principal line at the subpoint (projection of the principal axis onto the earth surface);

5. Rotation around $\underline{j}$ by the nadir angle of the principal axis, n: this brings the negative $\underline{i}$ axis up into coincidence with the principal axis, so that the coordinates are essentially the required camera-based system defined above;

6. Rotation around $\underline{i}$ (i.e., the principal axis) by the roll angle, ν.

The full transformation is applied to the cartesian components of the object in standard geographical coordinates:

$$\begin{Bmatrix} x' \\ y' \\ z' \end{Bmatrix} = \begin{Bmatrix} \cos\psi \cos\rho \\ \cos\psi \sin\rho \\ \sin\psi \end{Bmatrix} \tag{19}$$

It is useful to employ local geographic (geocentric) coordinates as an intermediate step, we denote these by (x, y, z), unprimed. We list convenient formal statements of the transformation:

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = [\,L\,] \begin{Bmatrix} x' \\ y' \\ z' \end{Bmatrix} \tag{20}$$

$$\begin{Bmatrix} a \\ b \\ c \end{Bmatrix} = [\,S\,] \begin{Bmatrix} x-r \\ y \\ z \end{Bmatrix} \tag{21}$$

$$\begin{Bmatrix} a \\ b \\ c \end{Bmatrix} = [\,S\,]\ [\,L\,] \begin{Bmatrix} x' \\ y' \\ z' \end{Bmatrix} - [\,S\,] \begin{Bmatrix} r \\ O \\ O \end{Bmatrix} \tag{22}$$

where

$$[L] = [2\,(-\gamma)]\,[3\,(\beta)] \tag{23a}$$

$$[S] = [1\,(\nu)]\,[2\,(\eta)]\,[1\,(\xi)] \tag{23b}$$

are the product rotation matrices.

In the grid program the calculations are planned so as to perform as much of the work as possible only once for each grid, leaving as little as possible to be calculated for each of the many points per grid. Thus, the transformation matrices of Equations (23a) and (23b) are completely evaluated at the start of a particular grid. Furthermore, along a meridian the longitude remains constant, so the program is arranged to preserve all products of transformation elements with functions of longitude, and to require only the insertion of new functions of latitude for new points along the meridian. A separate program section is written in addition to this, which provides the corresponding economy with parallels of latitude.

The calculation of the transformation elements is programmed in steps essentially as indicated by the above matrix expressions. In this way it is convenient to extract and store at various stages the partial transformations needed for the horizon, etc.

2.2.2 Validity of Image Points

The foregoing transformation to image coordinates includes points which actually do not appear in the picture. Tests are required to identify any such point on which the computer starts work. These tests are separate from those which reject an entire grid, although they may have that effect when the program is at the end of its list of possible points.

It is particularly important that all possible cases be detected because the present program must run unattended for periods of up to an hour, using grid parameters that are themselves being derived by the computer from more cryptic overall input data, so that it is virtually impossible for the operating personnel to anticipate the possible troubles of this sort.

One test (denoted "x-test") excludes points of object space that are hidden on the surface of earth beyond the horizon. In local geocentric coordinates the horizon lies in the plane $x = 1/r$ (earth radius of unity;

see section 2.3.2). The points to be excluded are those whose x coordinate is less than that value. This assumes that all trial points lie on earth, of course, for there are points elsewhere in space which are visible, yet fail this test. In practice this test is modified to exclude also a strip of earth surface near the horizon which is seen at too great a slant angle to provide useful cloud information; this is done by using an artificial x_{test} which exceeds the theoretical value of $1/r$. Empirical formulas have been used for simplicity:

$$x_{test} = 2/(r+1) \qquad \text{TIROS II} \qquad (24a)$$

$$x_{test} = (5/2) / \left[r + (3/2)\right] \qquad \text{TIROS III et seq.} \qquad (24b)$$

Their dependence on height (r, distance of satellite from earth center) is adequate for orbit ellipticity beyond that of TIROS V; the formulas in fact give reasonable results for r values as large as 2 or 3. However, it might be useful to derive a more rigorous expression to provide a fixed slant-angle limit for all r. In any case, however, the x-test is useful in combining both the horizon and the slant-angle tests.

The precise formulation of the x-test depends on programming details. If the local geocentric coordinates were normally obtained as an intermediate step, the x component would simply be tested then. The present program uses the scalar product between the unit vector to the subpoint and the unit vector to the point in question to derive x for the purpose of this test. Precision is not important here, so that one may take advantage of an arbitrarily curtailed multiplication command such as is available with the Bendix G-15D.

A separate test (denoted "a-test") is necessary to insure that the coordinate "a" of an object point be negative (in the camera coordinates a, b, c, of section 2.2.1). If this is not done, the mathematics permits full treatment, including plotting, for points on the wrong side of the camera, such that the alleged light ray comes from the object, <u>through</u> the focal plane (at the computed point), and <u>then</u> on to the lens. Although rare, this situation can be reached by erroneously calling for a grid when the nadir angle is near 180°, and occasionally it will occur in a correctly specified grid when the grid spacing is large and the nadir angle is large enough to make the focal plane intersect the earth (certain adjacent grid points then lie on opposite sides of focal plane).

It is also necessary to include an overflow test which, from a logical point of view, simply extends the region of space excluded by the preceding test. This test is placed at a convenient location following the divisions b/a and c/a (see sections 2.2.1). It is needed because under certain conditions the overflow generates completely erroneous focal plane coordinates which may elude the other point tests, such as the size test below. Viewed as an arithmetical test, this is very familiar in any programming. However, it also has the basic geometrical result of rejecting points lying near the focal plane, i.e., with very small "a" coordinate. It is believed that the program scaling avoids any other source of overflow.

A final test ("size test") merely limits the size of the output grid. After determing b/a and c/a, the sum of their squares is tested to insure that it is less than an arbitrarily assigned radius. Since this quantity is normally needed for correcting lens distortion, the test represents a trivial addition to the program execution time.

2.2.3 Lens and TV Distortion

The transformations of Section 2.2.1 assume zero distortion, i.e., the idealized straight-ray geometry of the pin-hole camera. However, any distortion which has been calibrated can be taken into account through a final transform which operates on the idealized image to produce the actual image.

The radial lens distortion of the TIROS lens can be described by an even polynomial function of the radius. The grid program calculates revised image plane coordinates (B, C) from the formulae:

$$w = (b/a)^2 + (c/a)^2 \qquad (25)$$

$$k = [\{(c_5 w + c_6) w + c_7\} w + c_8] w + c_9 \qquad (26)$$

$$B = kb/a, \quad C = kc/a \qquad (27)$$

in which the c_i are constants derived from an analysis of the lens calibration curve. No correction for tangential lens distortion has been found necessary for the TIROS work.

Further corrections could be made for the important additional distortions due to various units of the TV system, from camera through display. This would include non-radial effects found during calibration, as well as identifiable TV distortions which in general vary from one sequence of pictures to another. These would be known relative to the picture frame, so that the program would have to include the roll angle, which insures that the horizon, grids, etc., appear in the true orientation relative to scan lines and picture frame.

A common distortion, for example, is a simple stretch of the TIROS picture, either parallel to the scan or perpendicular to it. This is easily measured from the fiducial marks and normally it stays essentially constant throughout a sequence. With the roll angle properly included in the transformation coefficients of Equation (23b), this stretch could be incorporated easily into the plotting scale factors, and this distortion incorporated into the grids, with no change whatsoever in the grid point calculation program from the current version.

2.3 Special Picture Details

The predicted positions of several picture details are usually plotted in addition to the latitude-longitude grid lines. The basic purpose is to assist the analyst to achieve the proper alignment of the grid with the picture. In some cases these details also assist in interpreting the brightness of cloud images.

One type of picture detail is a landmark, i.e., a coastline which may be visible in the picture. Any such geographical feature can be plotted directly, using the transformation of section 2.2.1 on a stored sequence of latitude-longitude points. For fastest operation of the program, these would be stored as cartesian components in geographical coordinates (x', y', z') of section 2.2.1.

The following sections deal with picture details that are not known directly as latitude-longitude points. In general such a detail is first determined as an object orientation relative to the satellite, i.e., as azimuth and nadir angle. The image point then may be computed directly, or the corresponding latitude-longitude point for the effective object location may be determined to permit plotting the effective image on grids for a variety of camera orientations and locations. The latter procedure is used, for example, to indicate the path of principal points on each grid.

2.3.1 Transformation from Nadir and Azimuth

Let the location of a point be given by the nadir (n') and azimuth (ξ') angles of the line segment between it and the camera (primes distinguish these angles from the nadir and azimuth of the principal axis), and by the statement that the point lies on the earth surface. The distance from point to camera, D, is involved, but will not be known in general. These quantities are in fact spherical polar coordinates referred to the local coordinate system but with the camera lens as origin, the topocentric geographic coordinates of section 2.1.2, so we may write

$$\begin{aligned} (x - r) &= - D \cos n' \\ y &= - D \sin n' \sin \xi' \\ z &= D \sin n' \cos \xi' \end{aligned} \tag{28}$$

and the transformation of Equation (21) may be applied directly. The distance D in fact cancels from the ratios b/a, c/a, when only the direct image point is required, so that it may be set to unity at the outset for such calculations.

When the object latitude and longitude are required D must be calculated. For a spherical earth of unity radius, the basic geometry is shown in Figure 2, from which it can be seen that

$$D = r \cos n' - (1 - r^2 \sin^2 n')^{1/2} \tag{29}$$

Equation (28) may now be rewritten in the form

$$\begin{aligned} x &= r \sin^2 n' + \cos n' \, (1 - r^2 \sin^2 n')^{1/2} \\ y &= - (1 - x^2)^{1/2} \sin \xi' \\ z &= (1 - x^2)^{1/2} \cos \xi' \end{aligned} \tag{30}$$

The latitude, ψ, and longitude, ρ, are found by the transformation inverse to that of Equation (20):

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = [L]^{-1} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = [3(-\beta)][2(\gamma)] \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} \tag{31}$$

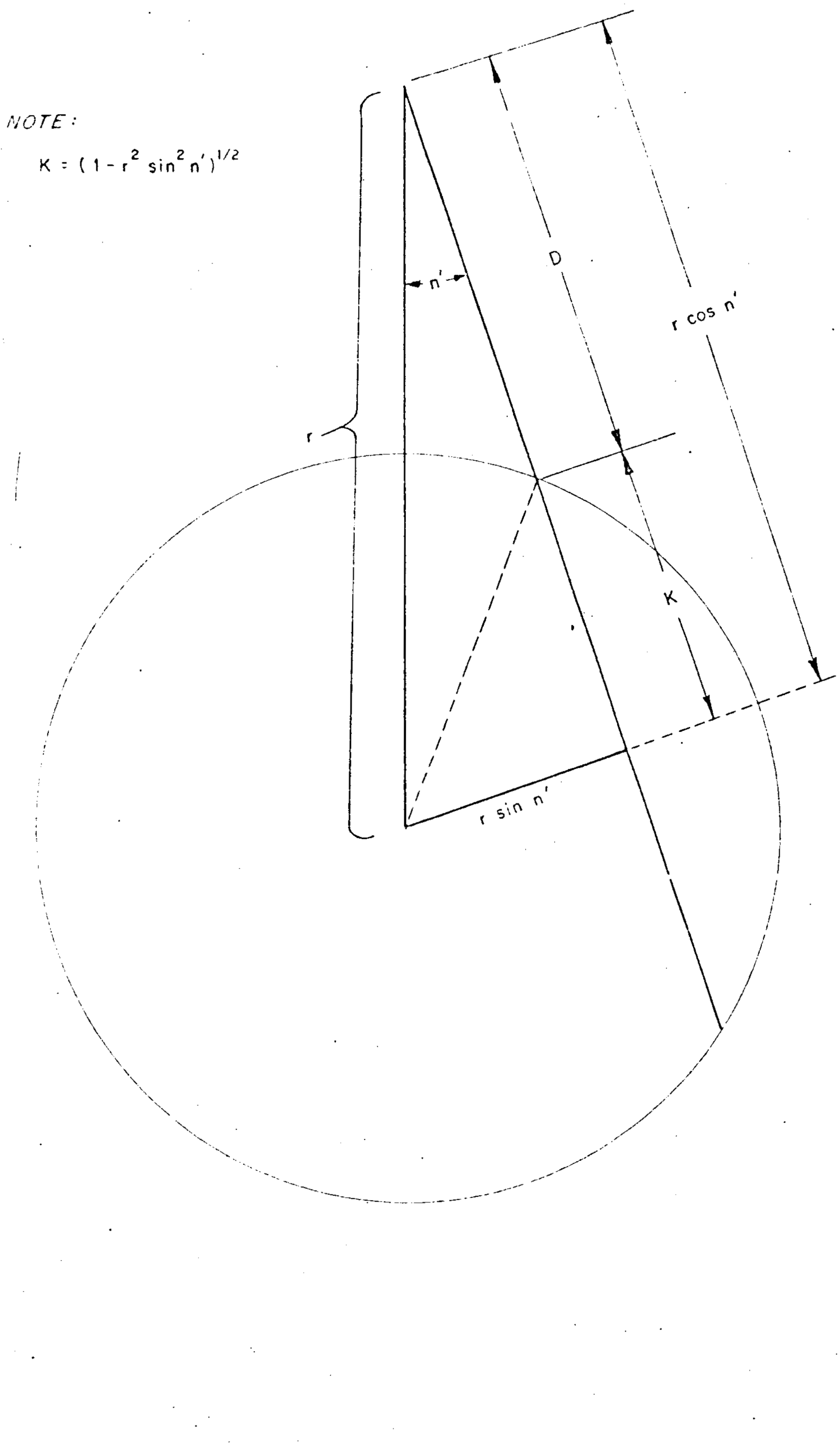

Figure 2 - Relation Between Object Distance, D and Object Nadir Angle, n

together with

$$\begin{aligned} \psi &= \arcsin z' \qquad -90^\circ \le \psi \le 90^\circ \\ \sin \rho &= y'/\cos\psi \\ \cos\rho &= x'/\cos\psi \end{aligned} \tag{32}$$

It should be noted that the storage of points derived for one grid, in order to plot those points on some other grid (such as is done for the path of principal points), is most speedily accomplished if the step of Equation (32) is omitted, and the points therefore plotted later using the direct transformation of Equation (22). This procedure avoids several trigonometric calculations as well as some additional multiplications.

2.3.2 The Horizon

It can be seen from Figure 2 that the nadir angle of a ray from the horizon to the camera satisfies the relation $r \sin n' = 1$. It follows from the first of Equations (30) that the horizon lies in the plane $x = 1/r$. The entire horizon can be traced out by supplying a sequence of values for ξ', and the image found by applying the transformation of Equations (28) and (21).

The horizon itself includes all values of ξ'. However, some portions of it may not be visible in the picture, this question depending on the angle between the principal axis and the ray from the horizon point. It can be seen that the horizon point nearest to the principal axis is characterized by an azimuth ξ' equal to the principal axis azimuth, ξ. Normally it is useful to start the program at this point, and plot the horizon in either direction until a point fails the output grid "size" test (section 2.2.2).

An alternate procedure is based on the observation that the horizon is completely symmetrical around the subpoint, so that actually what is involved is simply the azimuth difference, $(\xi' - \xi)$. One may in fact calculate special transformation matrices for Equation (21), in which ξ is arbitrarily taken as zero, and then in all cases start the program at the horizon point $\xi' = 0$. This is done in the present program, because for special reasons it deletes one multiply command per point and because the absence of TV distortion correction permits the program to calculate only half of the horizon, determining the remainder by reflection in the plane of mirror symmetry, (i.e., the points are stored, then replotted after changing the sign of E, the output from Equation (27)).

2.3.3 The Sun Line

The TIROS sun angle sensor purports to measure the roll angle of the camera and satellite around the principal axis. The sun angle is measured from the half-plane which is bounded by the principal axis and contains the sun, to a reference direction that is fixed in the satellite frame. The relation to the TV scan lines and picture frames apparently differs for the two cameras because of the way they are mounted, but this detail simply involves an additive constant that can be determined from engineering drawings, or perhaps even more quickly, empirically.

If it were available in time, and if it were reliable, the sun angle would be used to specify roll angle ν in deriving the latitude-longitude grids. Since it is neither, but some provision must be made, the grids may be drawn with an extra line (the "sun line") which extends from the principal point out through the image of the shadow of the satellite. The sun angle for a picture now specifies the angle between this sun line and a reference line which is fixed relative to the scan lines and picture frame details; the user merely rotates the grid relative to the picture to achieve this prescribed angle.

The nadir and azimuth angles of the satellite shadow are the quantities regarding the sun which are derived initially from the ephemeris. The image of the shadow can be calculated immediately from the transformation Equations (28), (21). In practice the shadow point image should certainly be plotted as an aid in detecting the extent to which apparent cloud brightness is being affected by Heiligenschein. In general, however, there are cases in which the shadow image would not appear even though the shadow line would be needed: these occur when the sun lies near the camera focal plane, or in front of it. In normal TIROS operations these situations will be avoided, but it is pointless to make the program depend on this. To remove this unnecessary restriction, Equation (21) is used to calculate only b and c, and the length $N = (b^2 + c^2)^{1/2}$ is used for normalization, rather than a. The sun line is plotted from the principal point to the point given in focal plane coordinates by

$$B = Db/N, \quad C = Dc/N \qquad (33)$$

where D is the arbitrarily chosen line length; at present this is the "grid size" test length. (Since the lens distortion is not applied here, the sunline extends somewhat farther out that any other grid feature.)

If the roll angle were used in the grid program, i.e., so that TV distortions could be included, the sun line would be deleted. Instead, the proper orientation of the grid would be given directly by having the computer draw the actual fiducial marks as they must appear in the picture.

2.3.4 The Sunglint Point

A TIROS picture shows with great brilliance any specular reflection of sunlight from a large open water surface toward the satellite ("sunglint"). We calculate the theoretical location of the sunglint point on the surface of the earth to plot its image in the latitude-longitude grid. This has two uses: it assists in orienting the grid relative to the picture whenever the sunglint does appear, and it assists in determining whether apparent cloud brightness has been affected by enhanced reflection at this angle when the sunglint point is not clear. It should be remembered that the theoretical point is not necessarily what the observer sees as the center of the phenomenon, both because surface effects enter, and because the geometry lacks up-to-down symmetry. However, with these warnings known, the calculated point is quite useful.

The sunglint azimuth angle is simply 180° different from that of the shadow-point. The shadow-point nadir angle gives the direction of the original rays of sunlight, before reflection. As seen in Figure 3, the idealized sunglint point is that for which the angle of incidence on earth $(N - \theta)$ equals the angle of reflection $(n_s + \theta)$; n_s is the nadir angle of the ray reflected into the camera, and θ is the central angle (at earth center) between the satellite subpoint and the point of reflection, i.e., the sunglint point. Thus

$$\begin{aligned} n_s + \theta &= N - \theta \\ \text{or} \qquad \theta &= (N/2) - (n_s/2) \end{aligned} \tag{34}$$

An additional relation serves to eliminate one of the two unknowns, n_s or θ; in its derivation we use unity earth radius, and r as the known satellite radial distance:

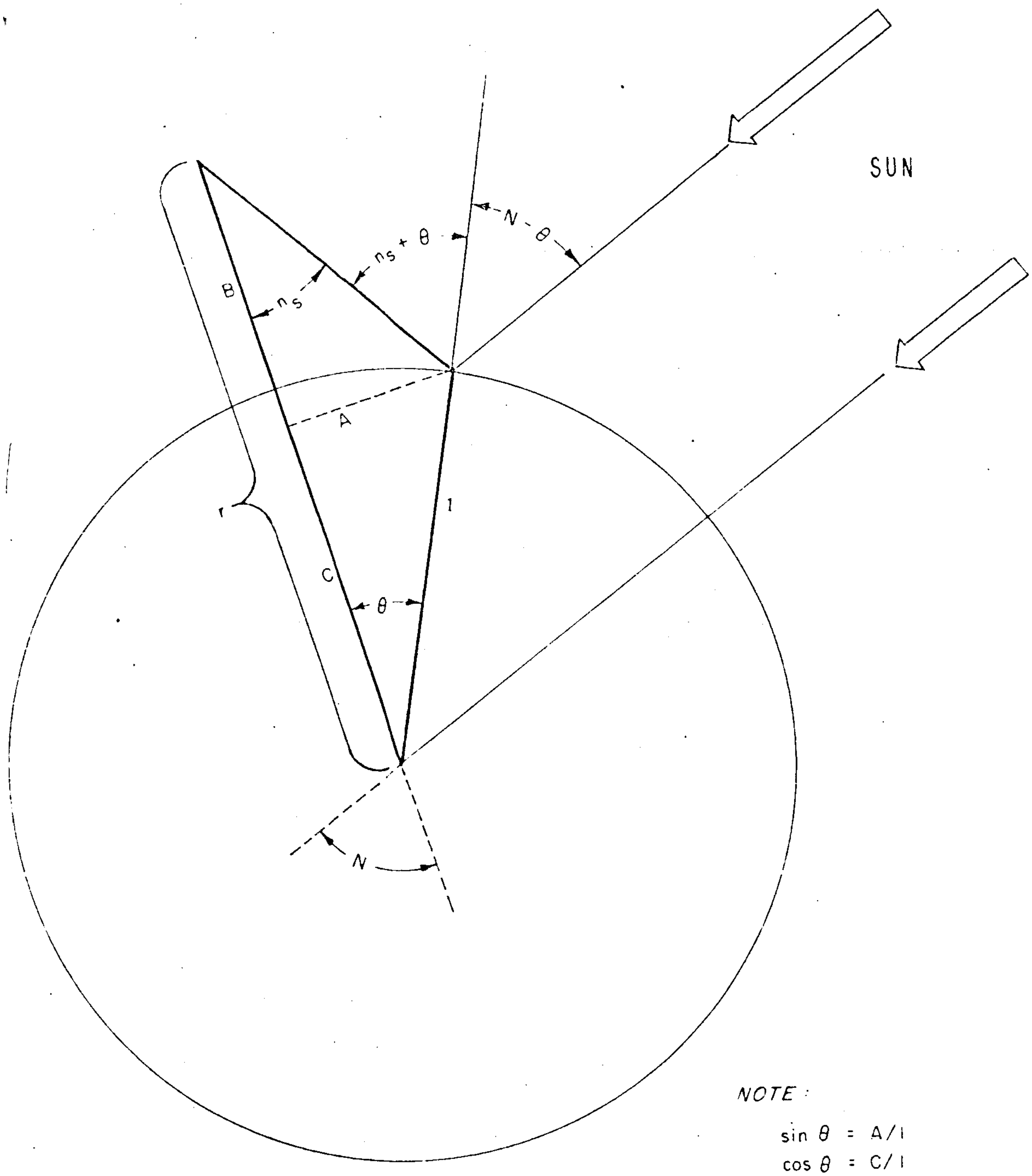

Figure 3 - Sunglint Geometry

$$\sin\theta = A$$
$$\cos\theta = C$$

thus

$$n_s = \operatorname{arctg}(A/B) = \operatorname{arctg}\left[A/(r - c)\right]$$
$$= \operatorname{arctg}\left[\sin\theta/(r - \cos\theta)\right] \tag{35}$$

These relations are combined into a form which is readily solved by iteration to find the nadir angle of the sunglint points:

$$\theta_{j+1} = (N/2) - (n_j/2) \tag{36a}$$
$$n_j = \operatorname{arctg}\left[\sin\theta_j/(r - \cos\theta_j)\right] \tag{36b}$$

The remainder of the calculations proceed as before, using Equations (28) and (21), etc.

3. Program Operation

3.1 Starting the Program

The grid program consists of two reels of paper tape and is stored on magnetic tape, unit #4, with a short loader tape. The current orbital elements and the sun data are stored on magnetic tape, unit #1. Figure 4 is a flow diagram of the program operation.

The program is started by loading the SPG paper tape program into the computer. The SPG program up-dates the orbital elements and the sun data to the time of nodal passage. These data as well as other data which are typed in are properly scaled and stored in drum memory in lines 13 and 18.

3.2 Typewriter Inputs

Typewriter input routines, specially written by ARACON personnel for the programs, are arranged for extreme convenience of routine use. Each input item is called for by a brief mnemonic output. After the mnemonic typeout, the computer operator must type in the requested information. Whole numbers, dates, hours, etc., are typed directly without the hollow period (the reload key). Decimal numbers require the use of the hollow period. Striking the hollow period in a whole number is immediately interpreted as an error, the computer takes control, a pair of tabs occur, and the number must be retyped. Similarly, if an extra hollow period is typed in a decimal number, a string of hollow periods serves to erase any typing error made before striking the release key. If, upon striking the release key, a hollow period has not been typed in an input supposed to be a decimal number, the type in error is entered.

The mnemonic typeouts are:

(1) Group I.

Orbital file number.

Sun data file number.

(2) Group II (Reference ascending node).

Orbit number.

Year.

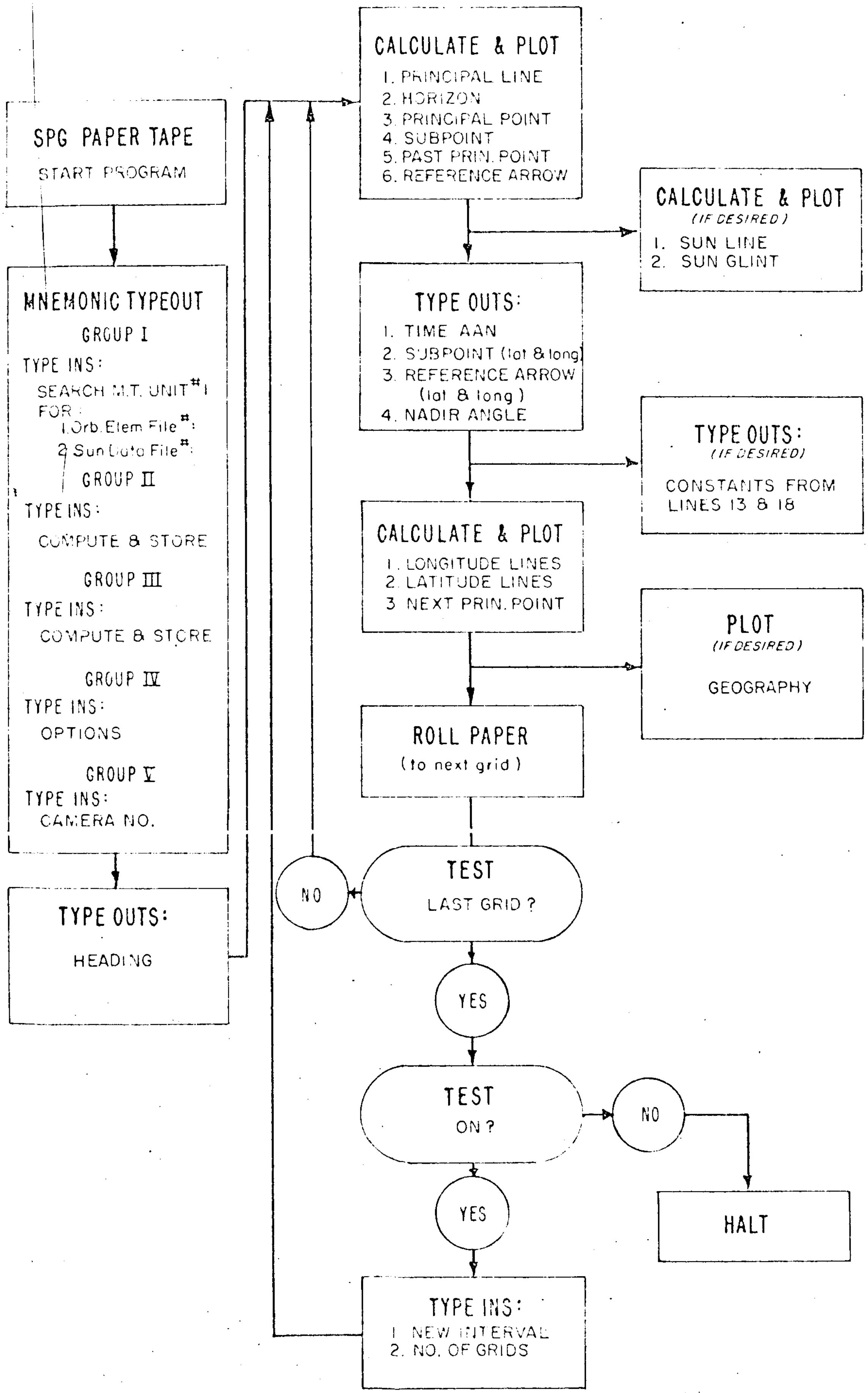

Figure 4 - Flow Diagram of the Grid Program Operation

Month.

Day.

GMT (HHMMSS).

East longitude, west is negative (DDDMM).

(3) Group III

GMT (HHMMSS) - reference first grid.

Day of first grid.

Number of grids to be plotted.

Time interval between grids.

Minimum nadir angle (NON).

Change per orbit in NON.

The angle along the orbit of track at which NON occurs (LOL).

Change per orbit in LOL.

(4) Group IV - Options.

Geography.

Current sunglint.

Sun line.

Stored constants.

(5) Group V.

Camera number. Camera number 1 is for the distorted lens, while camera number 2 is for no lens distortion.

Following the mnemonic typeouts and the required typeins, an alphanumeric heading is typed out for easy identification of the typeouts which accompany each grid made by the plotter. See Figure 5 for an example of the mnemonic typeouts, the typeins, and the alphanumeric heading. An explanation of the heading follows:

"T" is the time of the grid after ascending node.

"-SUBPT-" is the latitude and longitude of the sub-satellite point. West and south are negative.

"-ARROW-" is the latitude and longitude of the arrow which is plotted on each grid and serves as a reference point. West and south are negative. This arrow always points to the north.

"N" is the nadir angle for each grid plotted.

```
  FILE(V1000):V1000
  FILE:V1001

SS#:

     AB -
ORB:2538
YR:62
MO:12
DA:03
GMT(HHMMSS):20
ELONG(W IS -)(DDDMM):-125 0
1ST PIC -
GMT(HHMMSS):200700
PX:28

NO. OF PIX:3
TIME INTERVAL:100

LON:20.0
PERORB:6

LOL:24.8
PERORB:6

OPTIONS:  - 1   - 3   - 4   -

CAMERA  NO.?:  1

  T          -SUBPT-              -ARROW-          N

 7.00    22.1    -112.1      25.0    -115.0        20.2
 8.00    25.1    -110.0      25.0    -110.0        20.1
 9.00    25.1    -107.8      30.0    -110.0        20.

OK?:

NEW INTERVAL: 130

NO. OF PIXS: 1
 10.50    32.5    -104.2      35.0    -105.0        22.5
```

Figure 5 - The Mnemonic Typeouts and the Typewriter Inputs for a Grid Program

3.3 Plotting the Grid

After the alphanumeric heading is typed out, the computer calculates and plots the following for each grid:

(1) The principal line. This is a line segment along the principal axis and extends from the principal point to the horizon. It is the shortest distance from the principal point to the horizon.

(2) The horizon will then be plotted if it appears within the scope of the grid.

(3) The principal point is plotted as a small cross (+). This is the point of intersection of the principal axis with the earth's surface or with the celestial sphere if the principal point is not on the earth's surface.

(4) The past principal point is also plotted as a small cross (+). Since it is the principal point of the previously plotted grid, it will not appear on the first grid plotted, and, of course, it may or may not appear on any grid if the point is or is not geographically on the grid.

(5) The sub-satellite point - referred to as the subpoint - is plotted as a small square (□).

(6) The reference point is plotted as a small arrow (^). It is used to identify the latitude and longitude lines in the grid. It appears near the center of the grid and always points to the north. The numeric typeouts for each grid give the latitude and longitude of the point of this arrow.

The following options are available and will be plotted and/or typed out if requested: The option is requested by typing in a number following the mnemonic typeout, "options."

(1) The geography of western North America may be plotted if it appears geographically on the grid. This option is requested by typing in the number "1" following the mnemonic typeout.

(2) The current sunglint may be calculated and plotted as a small triangle (Δ). It is requested by typing in the number "3".

(3) The sun line may be calculated and plotted as a short line segment which extends to the horizon or an appropriate distance. It is requested by typing in the number "4".

(4) The constants which were computed and scaled at the beginning of the program and then stored in lines 13 and 18 may be typed out if requested. They are requested by typing in the number "5". This option was extremely helpful in debugging the program.

The calculation and/or plotting of the principal line, the horizon, the principal point, the subpoint, the past principal point, the reference point and also the plotting of the sun line, and the sunglint, if they were properly requested, are followed by an alphanumeric typeout. This typeout gives the time after ascending node, the latitude and longitude of the subpoint and the reference point, and the nadir angle for the grid which is being plotted at that time. After this typeout the longitude and latitude lines are calculated and plotted at five degree intervals. The principal point for the next grid is then calculated and it is plotted if it appears within the scope of this grid. A test is then made by the computer to determine if the geography option was requested. If the geography option was properly requested another test is made to ascertain whether the geography should be plotted on the grid. After the grid has been completely plotted, the plotter paper is then rolled forward and the plotter is then ready to plot the next grid. Figures 6 and 7 are examples of plotted grids with options. A test is then made to determine if the last requested grid has been plotted. If the last grid has not as yet been plotted, the sequence of operations is repeated and another grid is plotted. After the last grid has been plotted, the typeout "on?" occurs. At this point the operator may request any number of additional grids for any new interval desired.

3.4 Grid Size

The grid size is determined by one constant in the program. Figure 6 is an example of a grid with a 3 inch radius. This is the normal size plotted and is used by the weather bureau personnel to draw latitude-longitude lines directly onto the TIROS photographs. Figure 7 is an example of the same grid with a 5 inch radius. This is the size that is normally used with the TIROS film negatives and a photo enlarger.

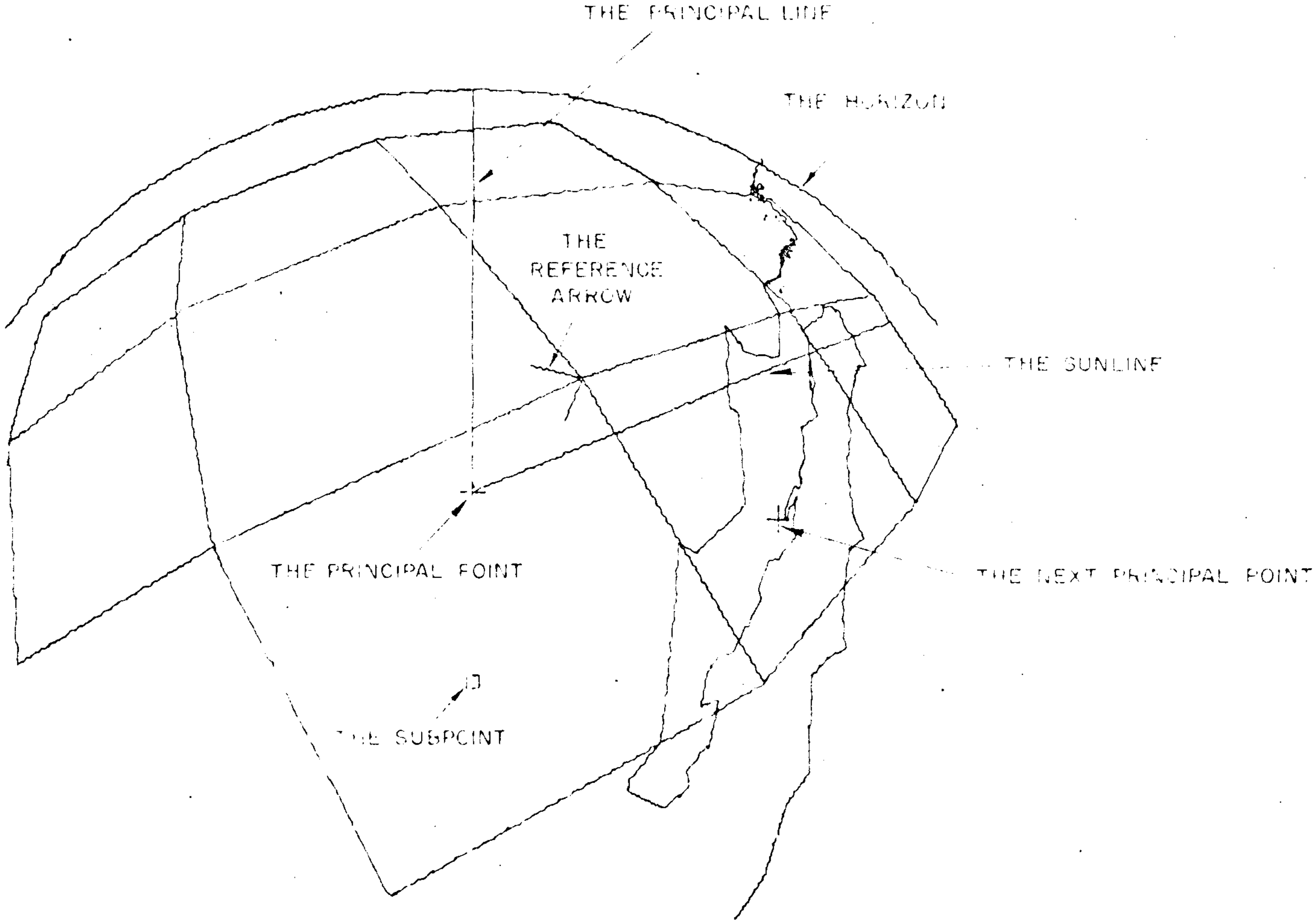

Figure 6 - A Three Inch Radius Grid

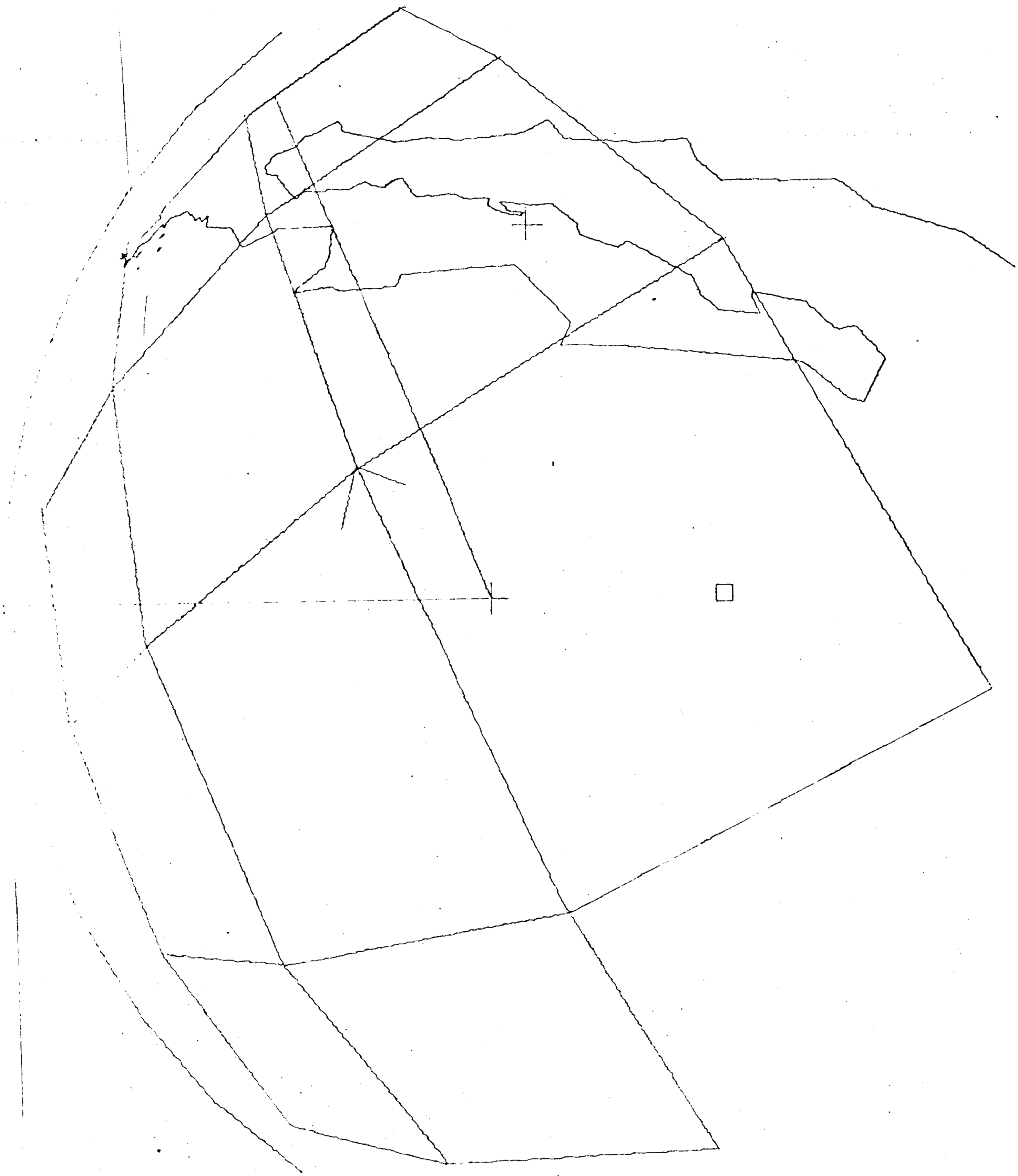

Figure 7 - A Five Inch Radius Grid

3.5 Time To Plot a Sequence of Grids

Standard procedure at the TIROS data acquisition sites is for the ARACON Computer operators to have the computer draw a set of grids in advance of each remote picture taking sequence. Grids are normally generated for one minute intervals for all the odd numbered frames. Since the cameras are set to take remote pictures at 30 second intervals for 16 minutes, it requires 16 grids at one minute intervals for a standard sequence of pictures. The average time for the operator to type in the required information and then for the computer to generate and plot the latitude-longitude sequence of 16 grids is about 60 minutes for an average of about 4 minutes per grid.

4. Program Organization

Three separate groups of calculations are involved in the grid program. Those in the first group are performed only once for an entire sequence of grids, those in the second group are performed once for each grid, and those in the third group are performed once for each point in a grid. The relative effect of these groups on the program timing can be estimated by noting that there are usually about sixteen grids per sequence, and the order of fifty to one hundred points in each grid (in the latter count one must include horizon points and at least a single border layer of rejected points, and every latitude-longitude intersection must be counted twice in the present program).

The first group includes:

1. all type-ins;

2. search and read-in operations for stored orbit elements and sun ephemerides;

3. updating of orbital elements, as described in Section 2.1.1, and their preparation and storage for use in the main program;

4. sun data interpolation, and transformation to orbital coordinates;

5. various bookkeeping operations concerning program options, typed-in grid specifications, etc.;

6. calculate time after ascending node at which perigee occurs, t_p, using the stationary orbit approximation of Section 2.1.3; that is, solve Equation 4b for M, given v, by iteration: $M_{i+1} = v - 2e \sin M_i$ where v is set equal to the argument of perigee, w, M_o is set equal to zero, and the iteration is continued until the change in M is satisfactorily small; then, $t_p = (T_n M)/(2\pi)$, essentially as in Equation 5.

The second group includes:

1. calculate satellite position in orbital coordinates from Equation 4a, b, 5, and 6, and the conversion to geographical coordinates by Equations 7 and 8.

2. calculate x_{test} by 24a or b.

3. calculate rotation matrices for 23a,

4. calculate orbital azimuth of North, Equation 17

5. calculate nadir and orbital azimuth angles from n_o, L_o values stored during first group operations, using Equations 15a, b, 16, for

(a) satellite principal axis

(b) shadow line from satellite,

and determine geographical azimuths by subtracting North azimuth according to step 2, above;

6. calculate rotation matrices for Equations 23b;

7. calculate product matrices specified by 23a, b, and supplementary transformation matrices for horizon calculations, Section 2. 3. 2, and for x-test, as described in Section 2. 2. 2;

8. test that the calculated principal axis nadir angle is small enough to warrant drawing the grid; if not, loop to next picture time according to type-ins.

9. determine on the basis of the nadir angle and satellite altitude whether the principal point lies on the earth and if it does, whether a useful first-grid-point can be derived from it;

10a. If the principal point lies on earth, find its latitude and longitude from Equations 30, 31, 32; in either case, set an appropriate indicator and store the data for use when plotting adjacent grids, so as to construct the required track of principal points;

10b. plot this principal point on the preceding grid; it will be noted that the preceding grid paper, and the complete set of transformation matrices for the preceding grid must still be present in the computer in order to accomplish this;

10c. if principal point is suitable for deriving a first-grid-point for the new grid, store it for this purpose separately from the storage of 10a, above; or, if principal point is not suitable, use a suitable fake nadir angle with Equations 30, 31, 32, and store the resulting fake principal point for deriving a first-grid-point;

11. calculate the sun-line coordinates, according to Equations 28, 21, and 33. Preferably the shadow point would be determined here and stored, and also plotted on the previous grid, to facilitate interpretation of apparent cloud brightness;

12. calculate sunglint point, Equations 36a, b, 28, 21; stepwise reduction of the trial interval should be used in the iteration of 36 a, b, rather than direct feed-back of the computed discrepancy, for both large and small values of N must be accommodated, and the convergence properties differ for the two cases; tests are performed first to determine whether sunglint is possible; the sunglint point also should be plotted on the preceding grid to facilitate use of sunglint-derived matchpoints as well as for brightness interpretation;

The preceding operations are performed prior to rolling the chart paper and prior to emptying the machine of the transformations for the previous grid, in order to permit the principal point track, etc., to be entered on it.

Other once-per-grid operations include:

13. calculate a first grid point by rounding the real or fake principal point to the nearest five degrees; calculate its transformation into the picture plane; if this point fails one of the validity tests, round the principal point in a different fashion, and calculate its transformation; this procedure is repeated if necessary to try all four grid points at the corners of the box containing the start-point; if none is accepted by the validity tests, the grid is rejected and the program goes on to the next grid.

14. if a start point succeeds, its latitude and longitude are printed out by the typewriter, and the program uses its transform, and the transforms of points with half-degree increments of latitude and longitude, to plot an arrow pointing north and with its tip at the startpoint.

The third group of operations may be considered as a single unit which accepts a value of latitude and longitude, calculates its transform into the picture plane according to Equations 19, 22, and 13, tests the validity of the point in all ways described in Section 2. 2. 2, applies the lens distortion according to Equations 25, 26, and 27 if the point is good, and then enters the subroutine modified for maximum speed by ARACON which moves the plotter pen along the required path to the new point. In the present program, various curtailed versions of this group are employed for certain applications in order to minimize operating time:

1. horizon points are calculated according to the alternate procedure of the last paragraph of 2. 3. 2; the transformation is somewhat shorter than the full matrix operation otherwise required.

2. Grid boundaries are found by incrementing from the start point along a constant longitude until a validity test is failed; the same occurs in the other direction along the same longitude, and then the latitude of the grid point nearest the middle of this zone is found. This is used to replace the original latitude of the start point. Next, the nearest to a central longitude is found by the same process, and substituted into the start point; then the nearest to central latitude is redetermined if the longitude changed; it is assumed that this process has sufficiently canvassed the area. Throughout this, of course, lens correction is deleted and no plot occurs.

3. In the present program the actual object-image transformation has been programmed in two different orders. One first performs the operations relating to the longitude of the point, so that only the remainder of the calculations is required for succeeding points along the line of that constant value of longitude. The other takes into account the latitude first, and thus is used for constructing parallels of latitude. Isolated points, such as sunglint, shadowpoint, subpoint, and principal points are found using the routine for longitude first, since that has slightly fewer multiply commands.

SECTION II

TIROS HORIZON SENSOR PROGRAM

1. Title

TIROS Horizon Sensor Program

Designation: H-1 or H-5

Computer: Bendix G-15D w/PR-2, PA-3

2. Purpose

Reduction of raw TIROS horizon sensor data to a form where it can be used to determine TIROS spin axis orientation in orbital coordinates.

3. General Principles

TIROS satellites are equipped with an infra-red horizon sensor mounted so that its optical axis is at an angle to the nominal spin axis of the satellite. For TIROS I thru IV, the angle is 90°; for TIROS V and VI, it is 70°. The angle of acceptance of the horizon sensor is of the order of 1°.

As the satellite spins, successive horizon encounters are marked by a rise and fall of the IR observed by the sensor. The IR intensity is telemetered to the ground during a part of the period when the satellite is within acquisition radius of the Command and Data Acquisition station. At the ground the signal is partially differentiated to enhance changes, clipped, and fed to a discriminator circuit that marks the time of horizon encounter.

The discriminator controls a clock counter that accumulates the number of milliseconds between horizon encounters. Upon encounter, the contents of the binary counter are transferred to a special-purpose computer that converts to decimal, codes the decimal number to teletype coding and inserts formats, and punches a teletype tape containing the appropriate material.

The fraction of a spin period occupied by earth or sky is related to the angle between the spin axis and the vertical, called the nadir angle. Let us setup coordinates at the satellite, x_1, y_1, z_1, where z_1 points down the spin axis, y_1 is in the plane containing the center of the earth and the spin axis, and x_1 is in the perpendicular. The cone swept out by the sensor axis is given by

$$z_1^2 \tan^2 \psi = x_1^2 + y_1^2$$

where

$$x_1 = z_1 \tan \psi \cos \sigma$$

$$y_1 = z_1 \tan \psi \sin \sigma$$

ψ is the cone angle

σ is the rotation angle with respect to the $y_1 z_1$ plane.

The surface of the earth, represented in parallel axes x_2, y_2, and z_2 is

$$x_2^2 + y_2^2 + z_2^2 = 1$$

where the radius of the earth is taken as unity. This coordinate system is related to the first by

$$x_2 = x_1$$

$$y_2 = y_1 - a \sin n$$

$$z_2 = z_1 + a \cos n$$

where n is the nadir angle, and a is the normalized distance between origins.

The formulae may be collected and all coordinate variables but z_1 eliminated. The mathematical condition for the intersection of the scan cone with the horizon is that the dixcriminant of z_1 vanish:

$$a^2 (1 - 2 \tan \psi \cos \sigma \tan n + \tan^2 \psi \cos^2 \sigma \tan^2 n)$$

$$- (1 + \tan^2 n) (a^2 - 1) (1 + 2 \tan^2 \psi \cos^2 \sigma) = 0$$

This can be solved for n, giving after some rearrangement:

$$n = \arctan \frac{\sin 2\psi \cos\sigma \pm 2\sqrt{(1 - \frac{1}{a^2})(\frac{1}{a^2} + \sin^2\psi \cos^2\sigma - \sin^2\psi)}}{2(1 - \frac{1}{a^2} - \sin^2\psi \cos^2\sigma)}$$

As the satellite moves in orbit, the nadir angle continually changes:

$$\cos n = \cos n_o \cos (L - L_o)$$

where n_o is the minimum nadir angle (the declination of the spin axis with respect to the orbital plane as equator), L_o is the argument of minimum nadir angle (right ascension in the orbital plane of the spin axis), and L is the argument of the position of the satellite.

Each interval between horizons (2σ or $2\pi - 2\sigma$) can be used to make an estimate of n. If these be plotted as a function of L, the plot can be used against a nomogram of n vs L for various values of n_o to find best estimates of n_o and L_o, the wanted attitude parameters.

4. Method

4.1 The data

The data is presented as a punched tape in teletype coding. The tape contains four digit groups representing (normally) the number of milliseconds between successive horizon encounters. Ten such groups constitute a line, terminated by a carriage return and line feed.

A group representing horizon-earth-horizon is followed by a space, while a horizon-space-horizon group is followed by a comma. Gaps in the record of more than 9999 ms are filled by 0000 groups, representing 10,000 ms.

Real time relationships are established by "clock start" and "clock stop" codings. These are designated by a sequence of a minimum of four 1000 groups followed by a minimum of three 0000 groups. The time of start and of stop, given in the preamble of the teletype tape, correspond to the end of the "1000" groups. Accordingly, clock start or stop is recognized 30,000 ms after it actually occurs. Recognition of clock stop is not really required, as the program can be stopped manually when the data runs out. It is obviously crucial, however, that the established starting format be present on the tape.

An assortment of errors, garbles, and non-data may occur in the tapes, requiring heavy editing by the program to permit maximum use of whatever good data may exist. The errors programmed against:

(1) The horizon discriminator was triggered by noise, or failed to recognize an actual horizon; or no signal was received due to fadeout or interruption for reception of other telemetry. These classes of erroneous data can be supressed by two tests.

(a) Successive groups must be followed by alternating spaces and commas. The second of a data pair thus disqualified is not used. It has another chance at qualification with the following group.

(b) The sum of two successive groups passing test (a) must be within 30 ms of one spin period of the satellite. The spin period, manually entered, may be determined by inspection of the data or from independent sources. The fate of a disqualified group is as above.

(2) Too many or too few digits in a group. This may originate in the tape preparation (the error there is almost invariably a dropped zero at the end of a group) or in the PR-2 tape reader attachment to the computer, where space or comma characters may be misinterpreted as digits, as well as digits misinterpreted. As a result, the program must be prepared to cope with a random number of apparent digits and groups in each line.

4.2 Computational Formulae

The time interval between successive horizon encounters is related to the nadir angle of the spin axis by

$$n = \arctan \frac{\sin 2\psi \cos \sigma \pm 2\sqrt{\left(1 - \frac{1}{a^2}\right)\left(\frac{1}{a^2} + \sin^2\psi \cos^2\sigma - \sin^2\psi\right)}}{2\left(1 - 1/a^2 - \sin^2\psi \cos^2\sigma\right)} \qquad (1)$$

where

n = the nadir angle

ψ = the angle between sensor axis and spin axis

a = geocentric radius to satellite in earth radii

$$\sigma = \frac{\tau_e}{2T} = \frac{T - \tau_s}{2T} \quad \text{(in circles)}$$

τ_e = the duration of an earth scan

τ_s = the duration of an adjacent space scan

$T = \tau_e + \tau_s$ = the rotation period of TIROS

Except for the singular case of $\psi = 90^{\circ}$ with the sensor axis at right angles to the spin axis, n is a two-valued function of σ, representing a true ambiguity resolvable only by the temporal behavior of n, which is not available to the computer.

The singular case, used for TIROS I - IV, reduces to

$$n = \arctan \frac{a^2 - 1}{1 - a^2 \sin^2 \sigma} \tag{2}$$

The sign ambiguity is not significant.

The nadir angle, as computed by one of the formulas above, is machine plotted as a function of L, the true anomaly of the satellite at the time of the observation of σ. For convenience, the plot is against ℓ, defined by

$$\ell = L - L_r \tag{3}$$

where L_r is the value of L at the time corresponding to clock start. For convenience, the starting point of the reckoning of L is displaced to ascending node, so that a quantity

$$L_s = L_r + W \tag{4}$$

is outputted, where W is the argument of perigee. The addition may of course result in more than 360°, in which case only the excess is outputted.

Other formulae:

$$L\ \text{(circles)} = \frac{t - t_p}{P_n} + \frac{\epsilon}{\pi} \sin \frac{t - t_p}{P_n} \tag{5}$$

where

t = time (GMT)

t_p = time of perigee (GMT)

P_n = nodal period of satellite

ϵ = eccentricity of orbit

then

$$L_s = \frac{t_s - t_p}{P_n} + \frac{\epsilon}{\pi} \frac{\sin t_s - t_p}{P_n} \qquad (6)$$

whence

$$\ell = L - L_s = \frac{t - t_s}{P_n} + \frac{\epsilon}{\pi} \sin (Q + \frac{t - t_s}{P_n} = \frac{\epsilon}{\pi} \sin Q \qquad (7)$$

where

$$Q = \frac{t_s - t_p}{P_n} = \frac{t_s - t_a}{P_n} - \frac{t_p - t_a}{P_n} \qquad (8)$$

with t_a = time of ascending node (GMT)

The last term of (3) is given by

$$\frac{t_p - t_a}{P_n} + \frac{\epsilon}{\pi} \sin \frac{t_p - t_a}{P_n} = W \qquad (9)$$

This must be solved by iteration. Fortunately, a simple cut-and-try procedure with W as the first estimate converges rapidly because of the small magnitude of ϵ for any reasonable orbit.

The geocentric radius of the satellite is given by

$$a = a_o \left[1 - \epsilon \cos (Q + \frac{t - t_s}{P_n}) \right] \qquad (10)$$

5. Program Organization

The program is loaded and checked from paper tape. It requires slightly greater memory capacity than the computer possesses, so that the typewriter input of non-repetitive material is accomplished first, the material is processed, and further program loading proceeds. The bulk of this initial computation occurs during input of subsequent data.

Typewriter input routines, specially written for this program, are arranged for extreme convenience of routine use. Each input item is called for by a brief mnemonic output. Whole numbers, dates, hours, etc. are typed directly without the use of the hollow period (the reload key). Decimal numbers require the use of the hollow period. Striking the hollow period in a whole number is immediately interpreted as an error, the computer takes control, a pair of tabs occurs, and the number must be retyped. Similarly, if an extra hollow period is typed in a decimal number, the typein error routine is entered. Accordingly, a string of hollow periods serves to erase any typing error made before striking the release key. If, upon striking the release key, a hollow period has not been typed in, an input supposed to be a decimal number, the typein error routine is entered.

As a further convenience and safeguard for the operator, a mnemonic output calls for a proof read at the end of a group of typeins. If an error is discovered, typing -1 (or any other negative number) sends the program back to the beginning of the group of typeins. Striking the release permits the program to proceed.

A pair of axes are plotted by the PA-3 plotter, preceded by a halt to permit adjustment of pen position. Another halt permits verification of data tape positioning in the PR-2 photoreader and setting of its control to "run".

A block of tape is read by the photoreader, corresponding to a line of teletype printing. A full line would consist of ten 4-digit groups separated by coded spaces or commas for a total of 50 characters plus the carriage return control, here interpreted as a stop. The space code is recorded as Bendix "W" (12) in the PR-2, while the comma becomes an "X" (13).

The capacity of the primary input buffer, line 23, being but 29 hex characters, an automatic reload occurs during input of a normal or near-normal data block. This results in 29 characters in the "bottom" of line 19 and an indeterminate number in line 23. A routine is required to normalize the remaining data in line 23 and graft it onto the first 29 characters, relocated more conveniently in memory.

Each character must now be examined in turn. If it is a decimal character, it is routed to its proper place in the binary conversion routine. If it is a W or X, it must be the fifth character; if it occurs early, zeroes must be inserted for the missing digits, and the bell sounds twice. The defective number cannot be rejected because of the necessity of maintaining the best possible estimate of elapsed time from clock start.

If the fifth character is a digit, it is ignored and the bell sounds three times. Fortunately, this fault is rare, as there is no apparent rational procedure to estimate time when spaces or commas are frequently missing.

If a group, now become a data word, is followed by a comma, it is made negative as a convenient way of flagging. This permits logical testing without masking.

Each converted group is examined to detect 0000 and 1000 groups. A count of 1000 groups is maintained. When any group other than a 1000 group occurs, the count is reset to zero. After the count exceeds 3, 0000 groups can be counted, both counts returning to zero at any break in the sequence. When the count of 0000 groups reaches 3, the clock is started with 30,000 ms if it is not already running, or stopped if it has been running. After clock stop the program can be rerun. Absence of clock stop coding in the data does not affect processing, but requires reloading the program for restart.

Once the clock is started, a 0000 group causes the immediate addition of 10,000 ms to the clock and a return to the data-word extraction routine is performed. Two registers are maintained, one for the previous word processed, the other for the current word. Tests are performed to assure opposition of signs of the words. Failure leads to addition of the absolute value of the current word to the clock, substitution for the previous word, and a return to data-word x extraction. The absolute values of the previous and current words are added, and tested to fall within 30 ms of the nominal spin period. Failure leads to the same route.

The surviving word becomes τ_e or τ_s, depending on flag, in the formulae of the preceding section.

The clock contains $t-t_s$, which is used to compute a, and later ℓ. a, with T_e, is used to compute n, which is entered with ℓ into a plotter routine which positions the pen to plot n against ℓ, marking a "$\pm$" in the appropriate location. Equation (1) admits of two values of n, both of which are plotted in sequence.

The plotter output, containing a large number of discrete scattered points, each representing a valid data word, is used with a nomographic overlay which permits selection of the most probable satellite attitude. The human eye can select the most consistent data points, rejecting the wild points caused by noise in the system (a few such points manage to survive the logical tests) as well as the orderly digressions from the proper curve caused by the horizon discriminator's pursuit of a cloud rather than the horizon itself.

6. Timing Considerations

Program time is not particularly dependent on computation time, but rather on edit time and plotter speed.

The internal organization of the G-15 D computer makes digit-at-a-time examination of randomly organized input data relatively difficult. A solution was found for 4-bit precession of data which permits computation during precession. Accordingly, much of the conversion, etc, is contained within edit time. The examination of each digit takes 5 drum revolutions or about .15 sec. A 5-character input word thus takes about 1 sec to process to the point of decision to plot. This is about 10 sec per 10 word teletype line. To this must be added about 3 sec per line for tape input and normalization.

Subsequent computations to the point of pen motion take about one second, straight-line coding being used in part to avoid time loss in subroutine entries. In spite of apparent greater formula complication of H-5 over H-1, the same subroutines are used so that total compute time is not greatly different.

Plotter speed is about 2 inches per second. The pen down-mark an "x" -pen up sequence takes 0.3 sec for each point plotted. Pen travel depends on data scatter, averaging perhaps 0.5 in between points for the H-1 (90°) program. For the H-5 (70°) program, alternate plotting of "true" and "false" values of n results in average pen travel per word of 5 in.

The number of "good" words plotted per orbit can be about 200. The total number processed may be 300.

The collected significant times are:

	H-1	H-5
	seconds	
Load	90	90
Set Plotter	30	30
Plot Axes	15	15
Typeins	60	60
Set Data Tape	30	30
Read and Edit 30 Lines	390	390
Compute 200 Points	200	200
Plot 200 Points	110	
Plot 200 Pairs		620
	925	1435

7. Other Computers

It is of interest to examine the characteristics of a computer that might facilitate these computations.

The edit function, which looms large in time consumption, would profit by an input system which can distribute each character to a separate word, or at least to a definite location independent of input length. Prompt subsequent access to such characters would telescope the edit and conversion process to a small fraction of its present length. A further possibility is character-by-character input, in which processing of each character would be complete before the next one is picked up.

Algebraic computation is not very significant in the total time, although gains here would be helpful.

Plotting time is significant in the H-5 program. It can be speeded by use of the faster (3" per second) Calcomp plotter. This plotter cannot be used with the Bendix G-15 D computer because of the limited speed of computation of control commands. ARACOM has devised a routine which computes and executes these commands at about 210 per second. There seems to be no method of further accelerating this process.

If the computer is capable of forming control characters significantly faster than they are required, and has output buffering capabilities useable with the plotter, then compute-while-plot is possible, reducing total time to near plot time.

Of the smaller-scale computers available, the CDC 160-A seems to conform to the above suggestions for features to accelerate program performance.

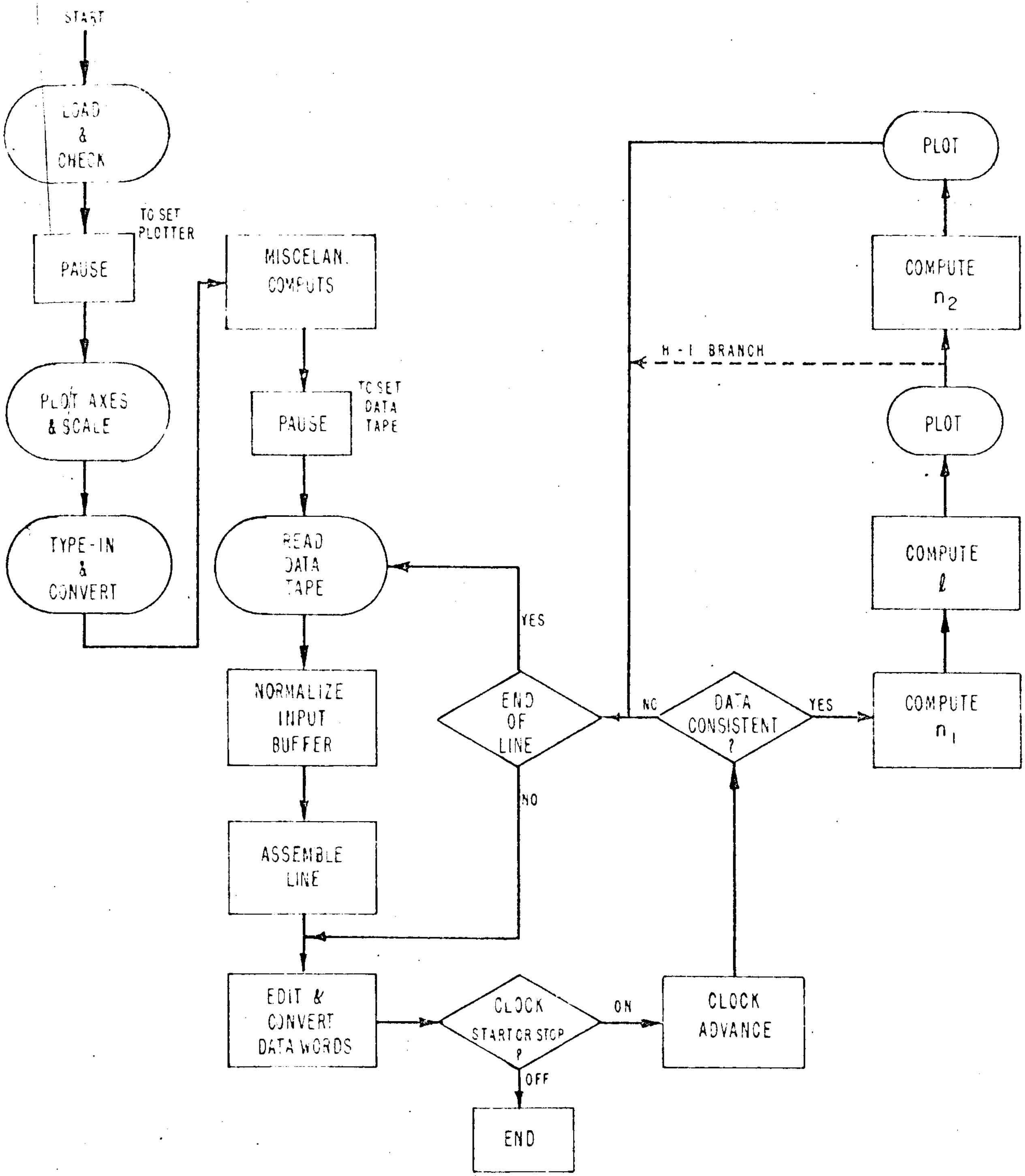

Figure 8 - TIROS H-5 Program Organization

SECTION III

COMPUTER REQUIREMENTS FOR EXTENDED TIROS OPERATIONS

1. The Bendix G-15 D

The use of Bendix G-15 D computers at the CDA stations originated with a unit available at Point Mugu, equipped with the then-new Calcomp plotter. A similar unit at ARACON Laboratories made possible program development. At the time of the original work, no other computer was known to offer the plotter as an on-line feature.

Accordingly, this computer has been used for a program that may be considered ambitious for its apparent capacity. The use of magnetic tape units, with the input buffering capacities of the Bendix has made possible reasonably efficient operation of this large program.

2. Program Time Distribution

The following is the approximate distribution by percentage of operating time of the various TIROS programs now used:

Picture Grids	70%
H-1 or H-5	8
MGAP	5
ODAM = OPAM	5
Subpoint and principal Point Lister	1
Sun Shadow	1
Miscellaneous Services	10
IR Gridding (Proposed)	(15)

3. New Computer Requirements

The scope of TIROS has grown far beyond the single-satellite, single-camera operation that originally was set up. Already the read-out stations have had to process daily operational data from three cameras (one on TIROS V and two on TIROS VI). According to projected launch schedules they will have to be able to deal with certainly four, and probably more, cameras in the near future. However, it has been found that operation with three cameras already strains the capacity of the Bendix G-15 D system with the present set of programs. With revised programming the system might eke out support for four cameras, but with no reserve for emergency work, and more than four cameras would be totally out of the question. Thus an increased computing facility is essential.

Among the many possibilities, it is believed that a change to the CDC 160A computer is probably the most desirable step. Several of its features appear ideal for the needs of the TIROS operations, and although individually these features are not unique to the 160A, it appears that the specific combination is unique at this price level. As described below, this computer offers only a moderate increase in general arithmetical speed over that of the present Bendix G-15 D system. However, the other features (input/output, etc.) combine to produce much more efficient system operation. Thus, for example, the grid program speed-up relative to the Bendix will be considerably greater than would be inferred by comparing only the arithmetical speeds.

One requirement for the new system, which is admirably fulfilled by the 160A, is flexible input/output performance with arbitrary codes on punched tape, and a good editing capability. These features are needed to permit conversion to automatic machine preparation and reading of teletype messages. With the increased scope of TIROS operations, the direct input of reperforated teletype data, and the direct preparation of summary messages (ODAM, OPAM, etc.) are most important from the human factors standpoint of eliminating errors and of smoothing out the operating procedures. From the computational standpoint this change is needed to make it feasible to read in lengthy tables of computed data as it is received from NASA computing center.

Since this is to be a replacement system in a daily operational enterprise, there should be no question of new and untried hardware designs, with possibility of protracted shut-downs for debugging. Thus an important feature of the 160A is the availability of the high-speed incremental plotter as a standard accessory, with thoroughly tested and proven CDC hardware linking it to the computer.

A requirement for efficient in-station use is the provision for full input/output buffering, with good interrupt hardware and commands, and it is important to insure that the plotter can be operated through the buffered channel. These requirements are met by the 160A. In suggesting a system below, we recommend at least one Auxiliary Memory Unit (of core storage); with this item it is believed that an additional parallel buffer channel in effect becomes available through careful programming, although we have not yet received firm engineering data on this matter. This would be most helpful in permitting normal input/output operations during grid plotting, or if more speed is needed later in facilitating the use of twin plotters.

It is most important that the new system be flexible enough to accommodate later revisions in TIROS operations. Here the 160A has two advantages: the basic unit suggested here is readily expanded in a variety of ways, and the computer is readily incorporated into larger complex of CDC computers such as is contemplated for NIMBUS operations. Thus it is unlikely that the 160A system would have to be replaced, along with all of the programs, in the way that it now seems necessary to replace the Bendix installation.

In addition to the foregoing hardware features, the CDC 160A software includes a translator which normally will produce an operating program in an interpretive language for the 160A (SICOM) directly from an already existing and debugged program in the Bendix G-15 D interpretive language (INTERCOM). This could be used to convert directly those operational programs which happen to have been left in Intercom, as well as to convert earlier versions of the other programs whose logic and basic algebra were initially tested in Intercom. For permanent programs it would be preferable to rewrite the entire library now used on the Bendix for TIROS, but as a back-up during initial operations with the new system the translator feature could be very helpful.

Another software feature of the 160A is a FORTRAN compiler. We emphatically do <u>not</u> recommend that the regular operational programs be produced with FORTRAN for the present work. However, again this is a feature which might be useful during initial operations with the system, by producing usable interim programs with least effort. Unfortunately the compiler requires magnetic tape units, which we do not recommend for the TIROS operations, but the compiling might be accomplished during rented time on other 160A installations.

A final software argument in favor of the CDC 160A is the consequence of the adoption of this machine by various groups at NASA for use as a computer-to-teletype buffer. An extensive set of programs has been built up to handle the necessary input/output operations, as well as to accomplish the relatively complex editing that is required. For example, it is understood that programs already exist for dealing with the data transmissions from NASA computing center, which were mentioned above. These programs would be a tremendous advantage in setting the new system into efficient operation.

4. Performance Data, CDC 160A

As is the case with several modern lower-priced computers, much arithmetical hardware has been omitted from the CDC 160A. Also, the basic word size is so small (12 bits) that multiple precision operations are needed for most work. Consequently, relatively extensive software is required for normal operations, and the performance of the available arithmetical subroutines is most important in determining the overall speed of the computer for normal scientific applications.

According to the CDC 160A Program Catalog 4 (November, 1962), only fixed-point arithmetic subroutines are available for use in machine-code programs. A set of subroutines provides double-precision fractional arithmetic (22 bits), which is probably adequate precision for our purposes. No higher-precision subroutines are available for machine-coded programs. For comparison we list the speed of the Bendix G-15 D for optimum-access coding, which is representative of our programs in the parts which are repeated most often, and the CDC 924, which is the next larger machine (times are in milliseconds):

	CDC 924 (23 bit)	CDC 160A (22 bit)	Bendix G-15 D (28 bit)	Bendix G-15 D (15 bit)
Add	.010	0.225	0.6	----
Sub.	.010	0.300	0.6	----
Mult.	.038	5.0 to 10.0	20.0	10.0
Div.	.038	10.0	20.0	10.0

As this table shows, the performance of the 160A is not particularly impressive for machine-language coded arithmetic. In addition, it has the disadvantage of lacking hardware for overflow testing, and it has only a single-precision register for the shift operations required in scaling. These operations therefore require additional programming. It would seem that the 160A is not well suited to machine-language computations, and it appears that this is tacitly admitted by CDC, for according to Program Catalog 4, they offer no subroutines to calculate standard functions other than a curious "9-bit quick sine" which is of no use to us.

On the other hand, relatively good performance is shown by each of the three interpretive languages offered for the 160A, in that the additional complications of triple or quadruple precision floating-point calculations, with programmed overflow tests, are handled with relatively little increased operating time over that needed for the machine-language operations quoted above. Part of the reason, of course, is that essentially no penalty is paid for using an interpretive system when the act of "interpreting" requires only a handful of fast machine logic as contrasted to the very large number of machine logic commands which are used to accomplish the simplest arithmetical operation.

The three different interpretive systems offered for the 160A are SICOM (almost identical to Bendix G-15 D Intercom, and used primarily with programs translated automatically from Intercom), INTERFOR (similar to SICOM but with important advantages in index register treatment and in looping and jump commands), and a FORTRAN Interpreter which apparently is designed specifically to have the most desirable characteristics for a FORTRAN Object-Code Language. Essentially a full complement of function subroutines is available in each language. The speeds given here (in milliseconds) are to be compared with the previous table of machine-language speeds:

	FORTRAN (------)	SICOM (10 decimal digits)	INTERFOR (33 bit)
Add/ Sub.	4.0	1.0 to 2.0	0.85
Mult.	14.0	6.0 min 22.0 max	14.0
Div.	19.0	9.0 min 30.0 max	25.0

In addition to superior logic commands, INTERFOR has the advantage over SICOM and FORTRAN of allowing very easy transition back and forth between the standard INTERFOR Language and direct machine language. Thus one may use the excellent high speed editing and general data processing capabilities of the 160A in addition to having the convenience of a flexible and fast interpreter. This provision in fact is quite complete: the INTERFOR package carefully protects the word locations addressed by the Interrupt hardware. An assembly program with symbolic addressing, relocatable output, etc., is available for INTERFOR.

According to local information the only FORTRAN for the 160A is one that has been adapted from the CDC 160 program; another FORTRAN is due for release soon by CDC, but is not yet available. In any event the CDC FORTRAN literature explicitly states that "...programs which have been compiled independently cannot be linked together at execute time." Precisely how limiting this really is, is not clear, but it does seem clear that the system is not convenient for producing blocks of programming for assembly into a variety of larger programs in the way that our TIROS program library is built.

5. Suggested CDC 160A Configuration

The following is a suggested configuration suitable for handling up to 5 TIROS cameras taking pictures on a schedule similar to that now obtaining, while retaining some reserve capacity for IR gridding. It is readily expandable for even more cameras.

160A	Base Unit	$90,000
161	Typewriter	10,500
365	Plotter (300 steps per sec)	~ 9,000
169-1	Auxilliary 8K Memory	50,000
	or	
2-606	Tape handlers and Control Unit	------

Expansion would call for an additional plotter and possible special hardware to assure optimum speed of both plotters without undue program constraints.

It should be noted that the CDC-924 base unit with typewriter and plotter offers the same effective storage as the 160A with additional 8K memory unit. Its effective computation speed is essentially 100 times that of the 160A (see table above). Additional cost is about 25 percent. Addition of tape units would move the cost margin much narrower.

REFERENCES

1. Glaser, A. H., et al TIROS Meteorology, Final Report, Contract AF 19(604)-5581, AFCRL 613, Allied Research Associates, Inc., 1961.